

Neural Network Ancillaries For Wind Energy Forecasting

Mohamed A. El-Sharkawi

Computational Intelligence Applications (CIA) Lab.

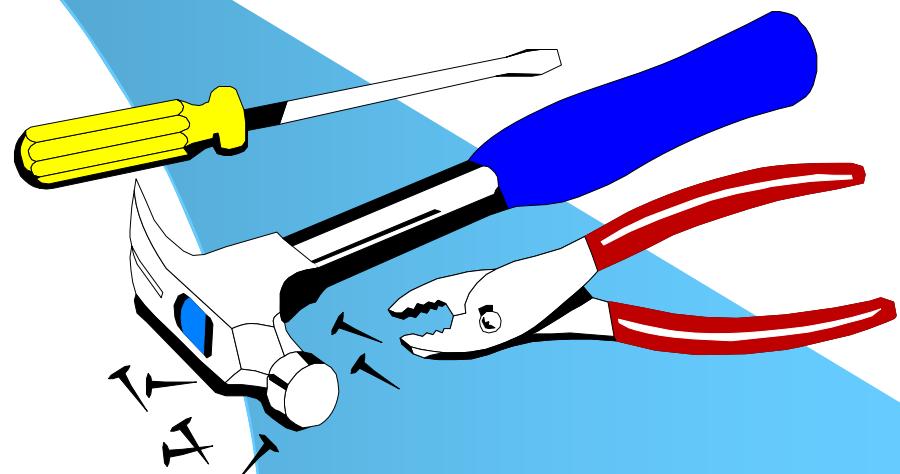
Department of EE, Box 352500

University of Washington

Seattle, WA 98195-2500

elsharkawi@ee.washington.edu

<http://cialab.ee.washington.edu>

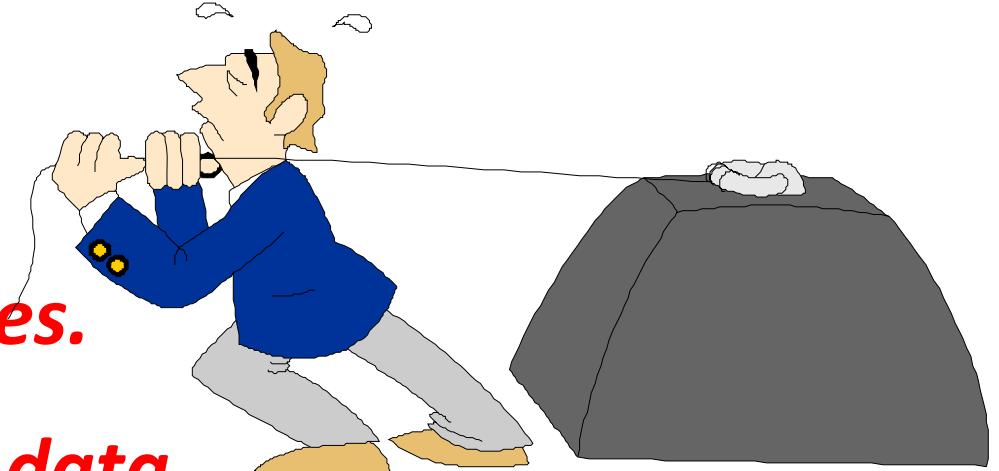


Why NN?

- Complex nonlinear mapping through a set of input/output examples.
- No structured model.
- Variables can be easily include or excluded.
- Superior noise rejection capability.
- Fast executions.

NN Challenges

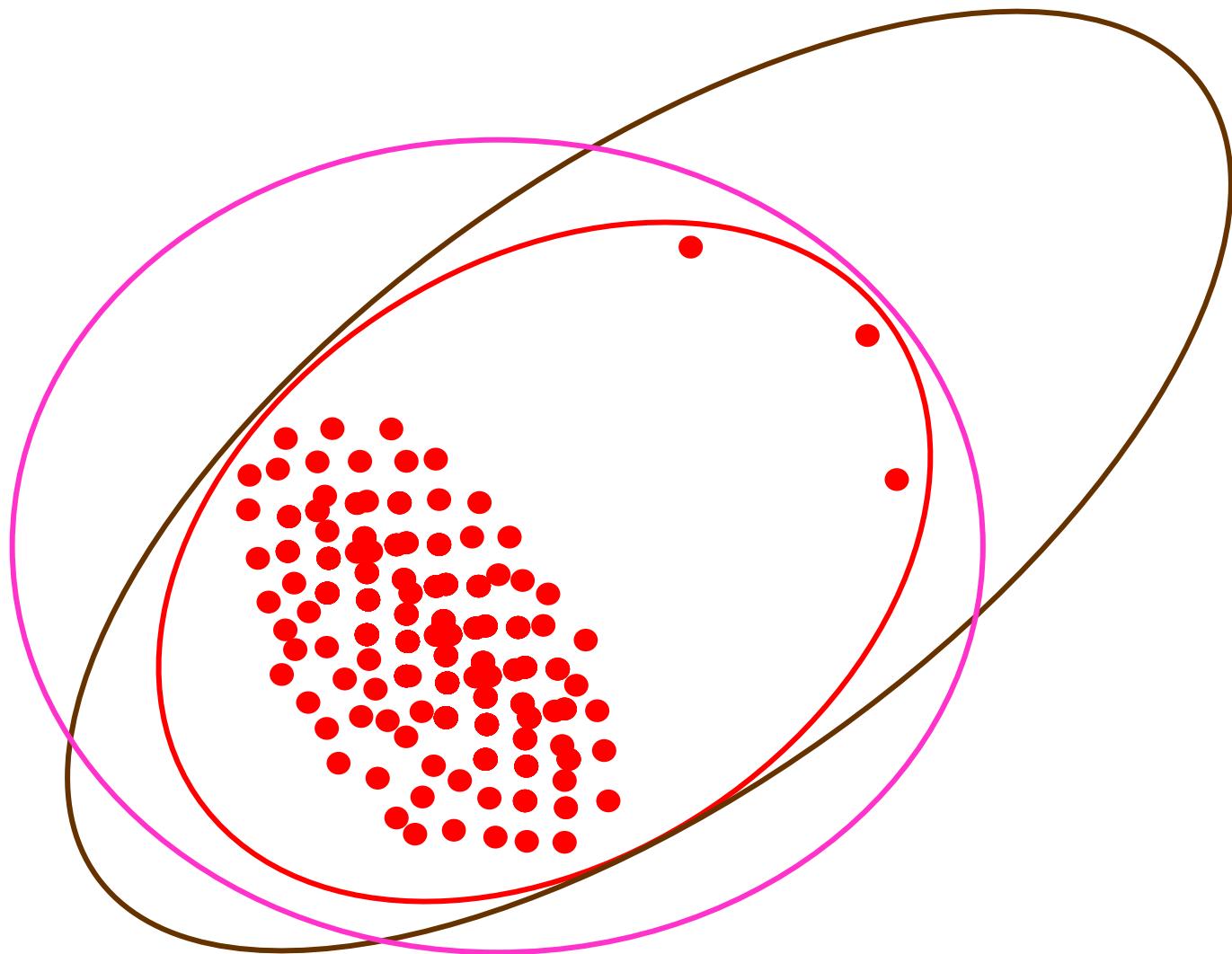
- NN architecture*
- Correlated Features.*
- Range of training data*
- Data spanning in the operational space*
- Data statistical properties.*



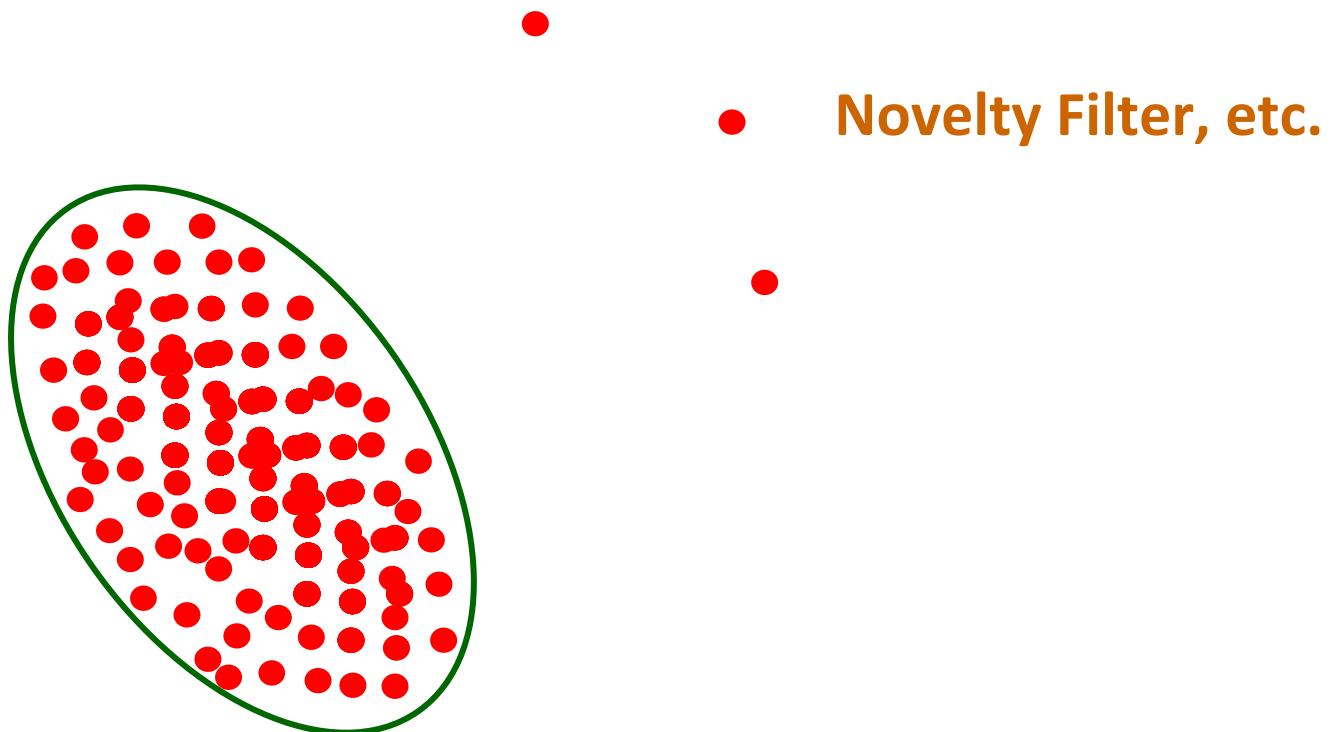
NN Challenges

- *Sensitivity to variation in topology, operation, and control characteristics.*
- *Memorization.*
- *Saturation.*
- *Adaptation to system dynamics.*
- *Learning protocol*

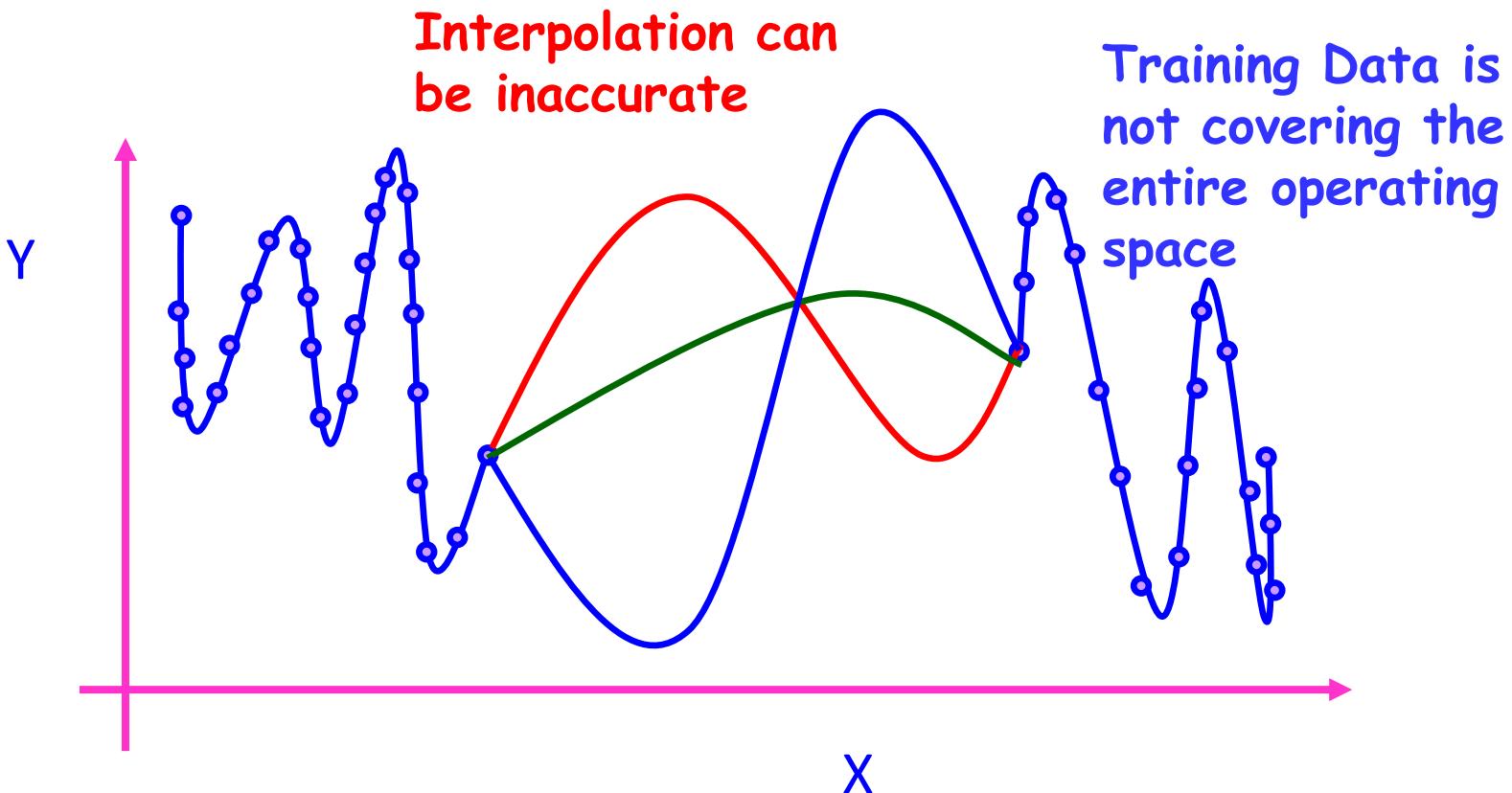
Range of Training Data



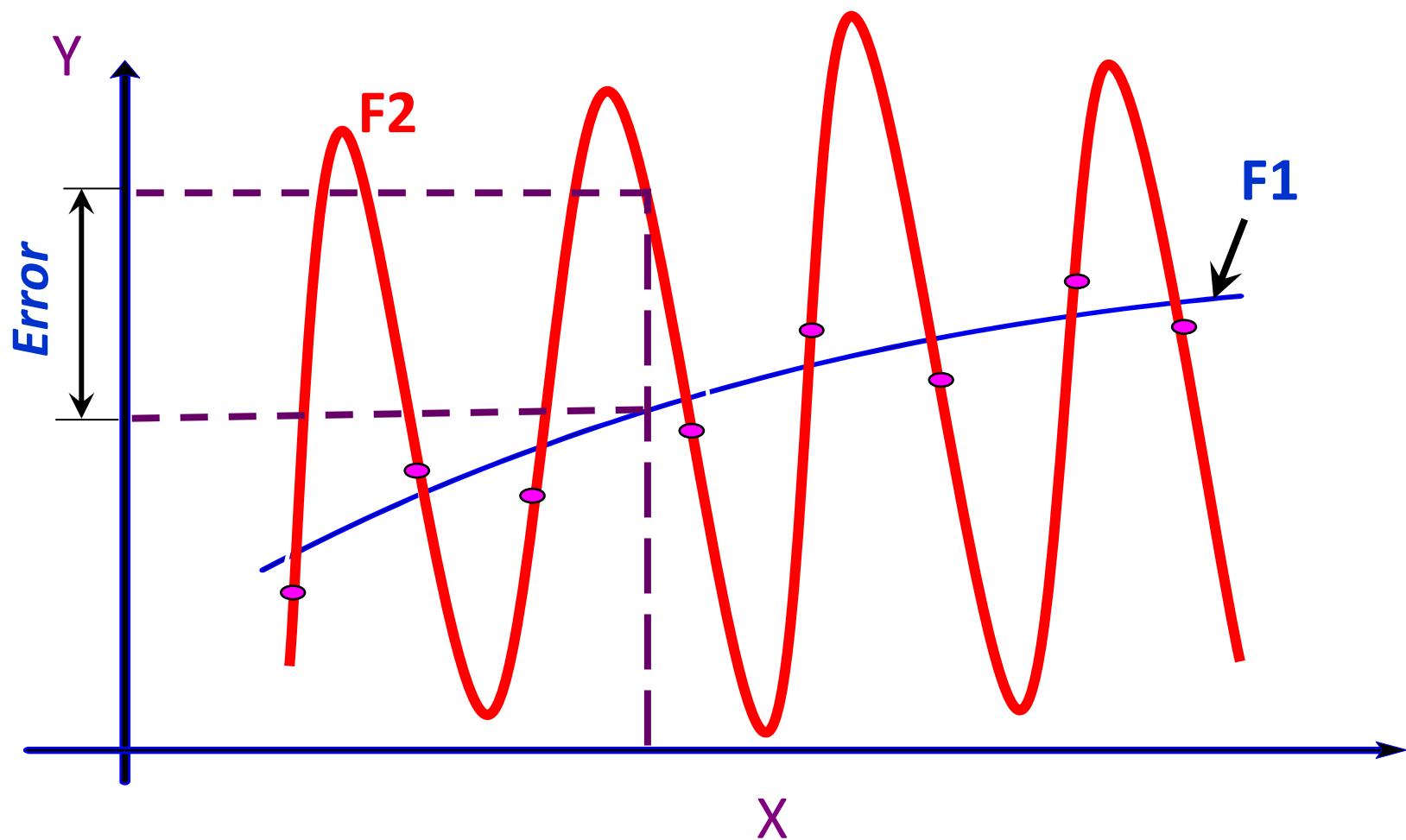
Range Limited Training

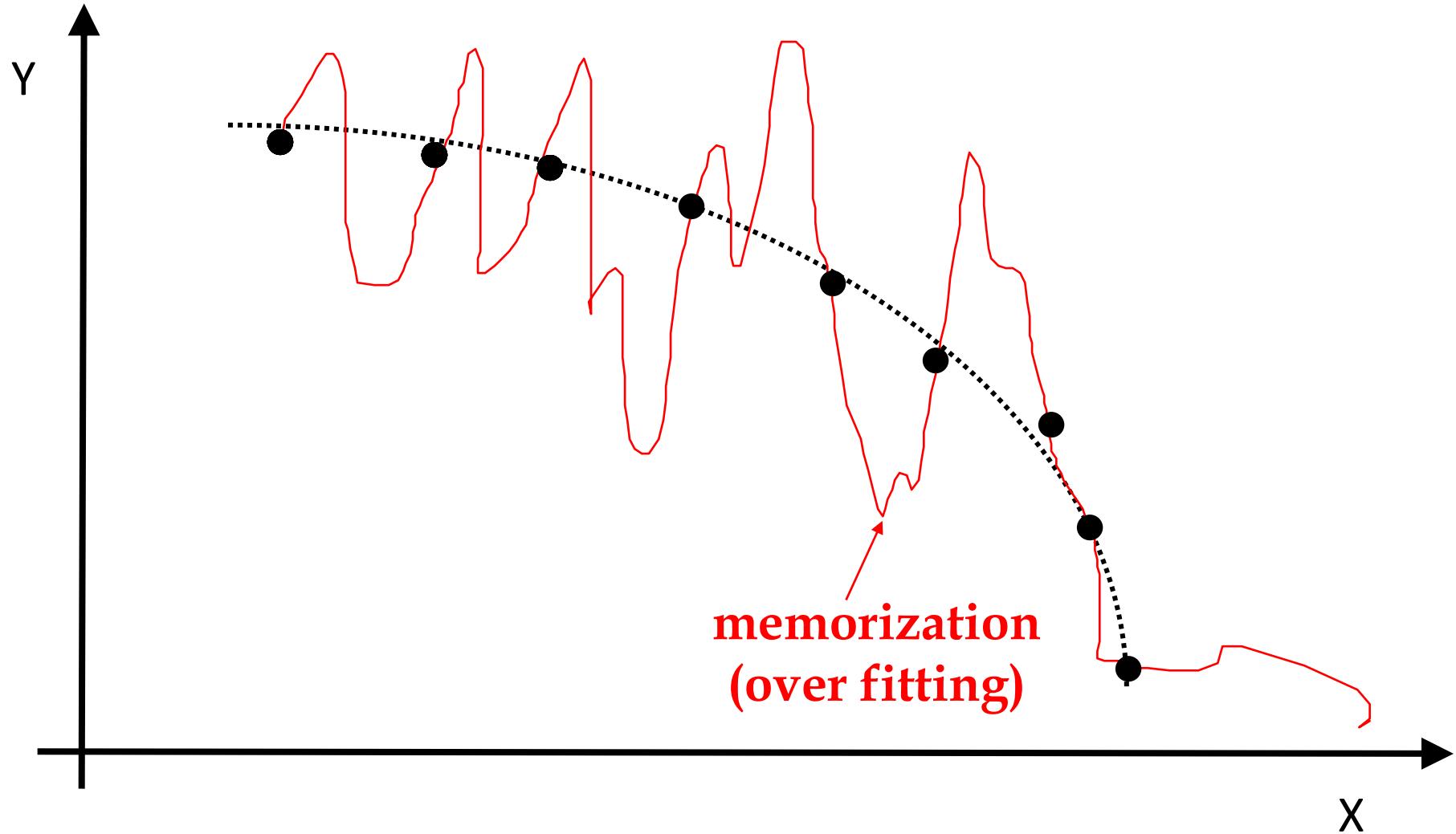


Distribution of Training Data in the Operating Space

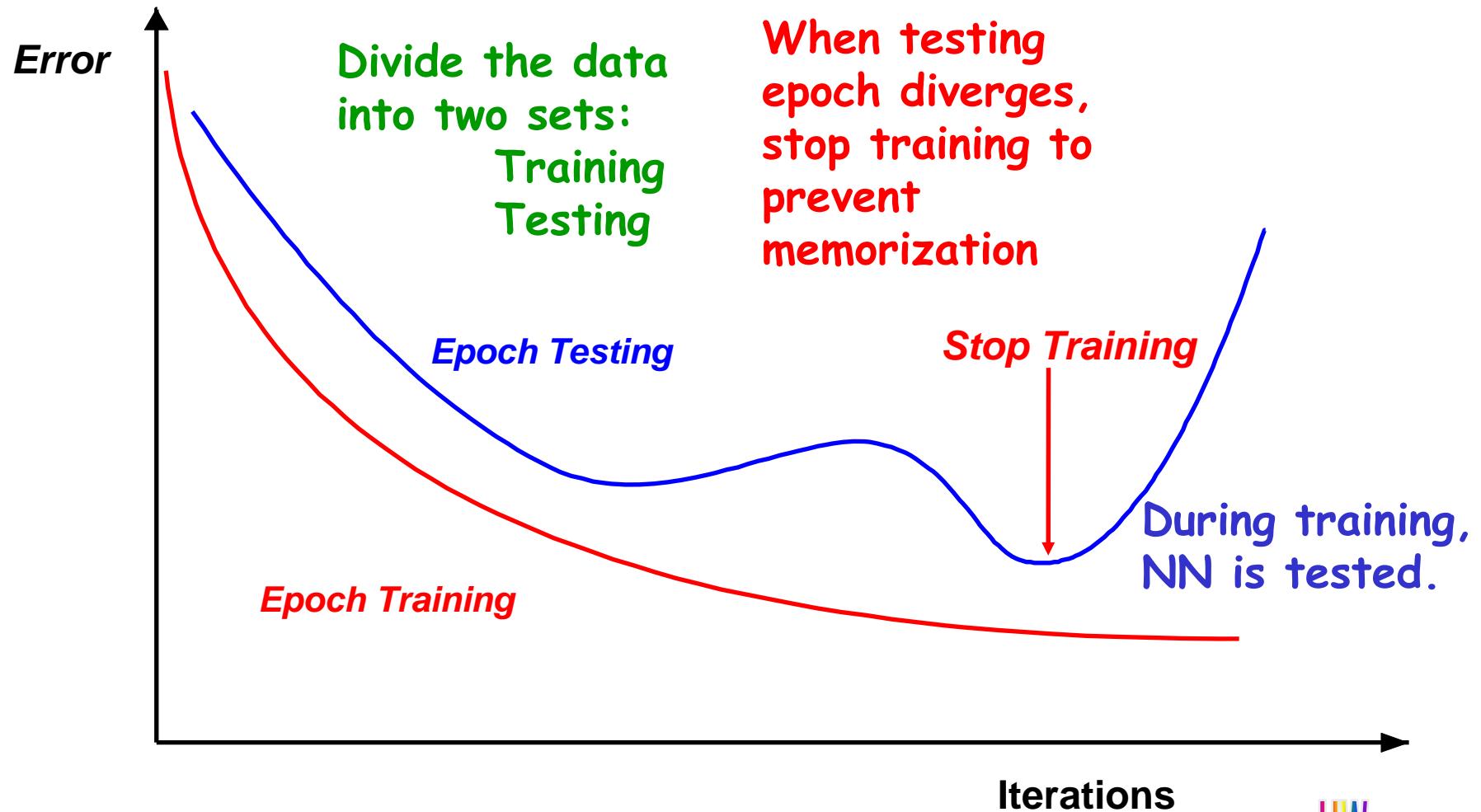


Learning versus Memorization



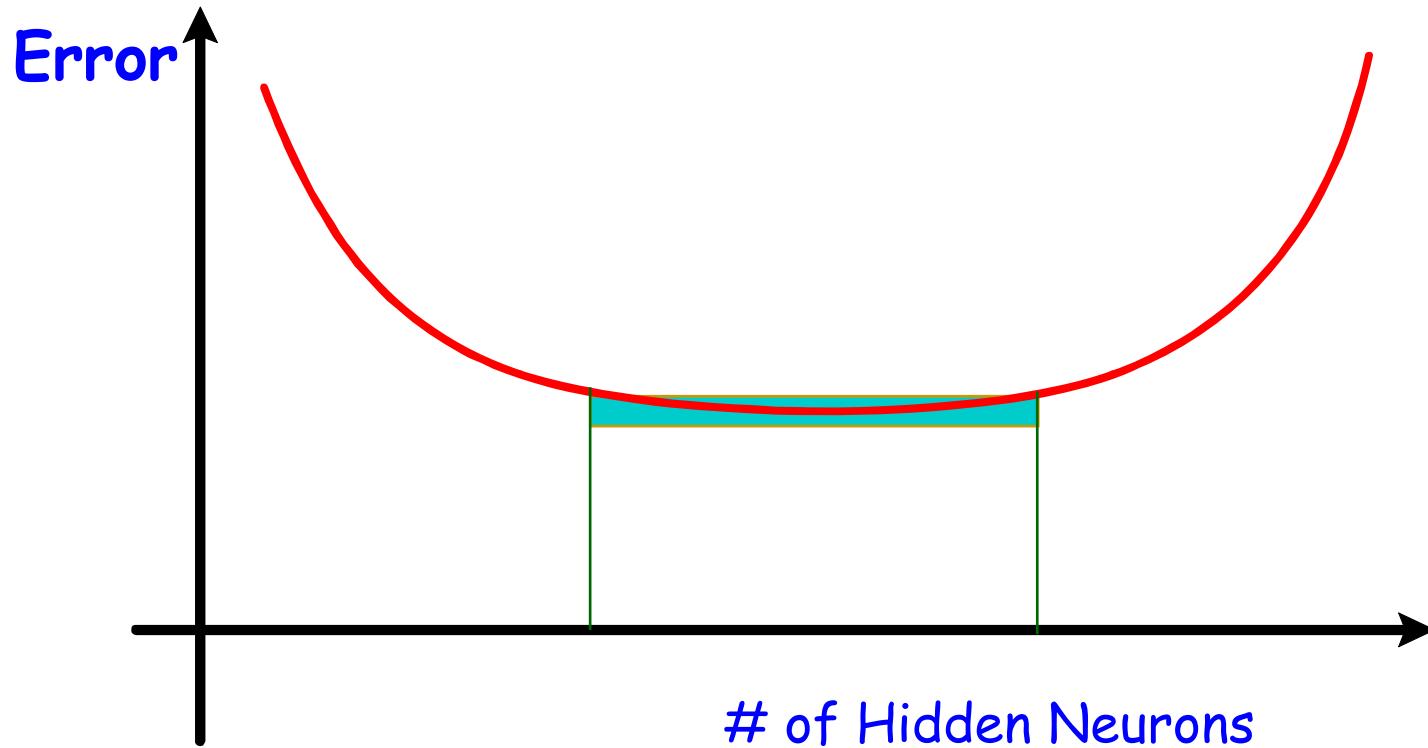


Cross Validation



Neural Net size

- The number of hidden neurons, must be matched to the complexity of the classification boundary.
- Cross validation among neural networks can provide good approximation of the net size.
- The best NN structure is where the error of the NN is relatively unchanged.



Data Normalization



Range Normalization

	<i>Feature 1</i>	<i>Feature 2</i>	<i>Feature k</i>
<i>Pattern 1</i>	x_{11}	x_{12}	x_{1k}
:	:	:	:	:
<i>Pattern n</i>	x_{n1}	x_{n2}		x_{nk}

$$Max_1 = \max(x_{11} : x_{n1})$$

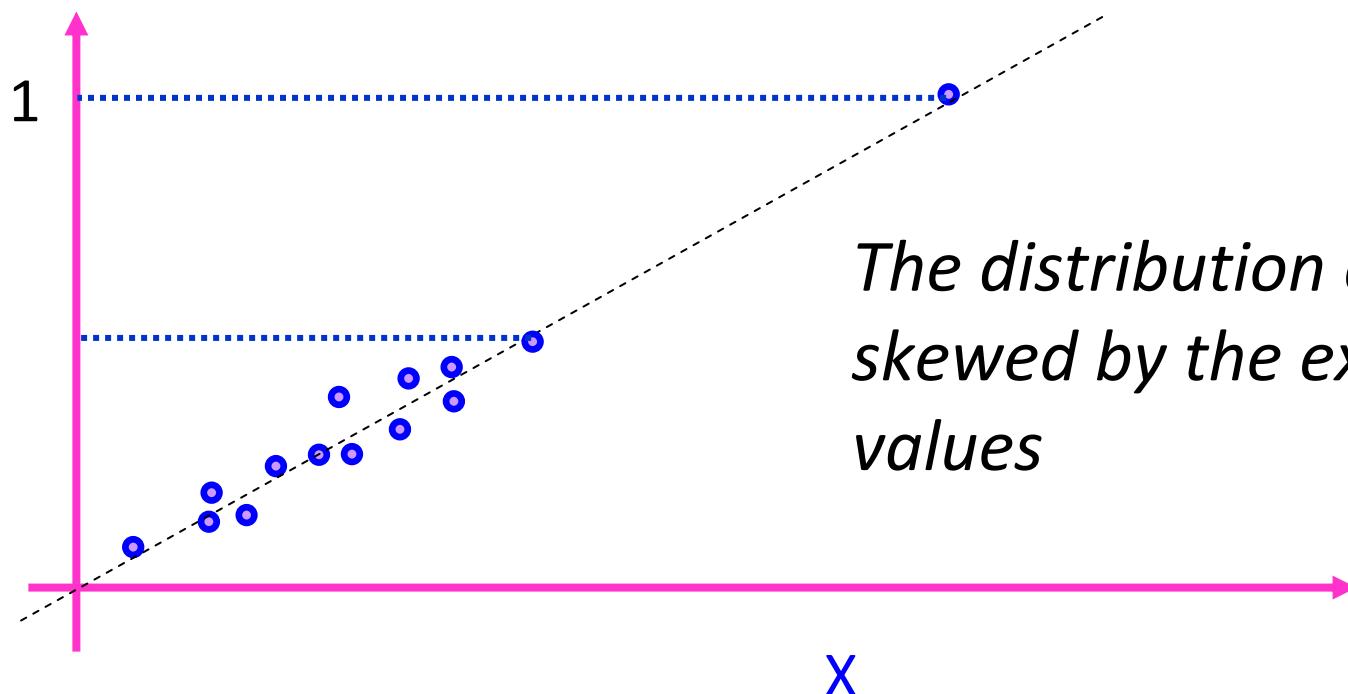
$$Min_1 = \min(x_{11} : x_{n1})$$

Range Normalized Input

	<i>Feature 1</i>	<i>Feature 2</i>	<i>Feature k</i>
<i>Pattern 1</i>	$\frac{x_{11} - Min_1}{Max_1 - Min_1}$		$\frac{x_{1k} - Min_k}{Max_k - Min_k}$
:	:	:	:	:
<i>Pattern n</i>	$\frac{x_{n1} - Min_1}{Max_1 - Min_1}$			$\frac{x_{nk} - Min_k}{Max_k - Min_k}$

Problems with Range Normalization

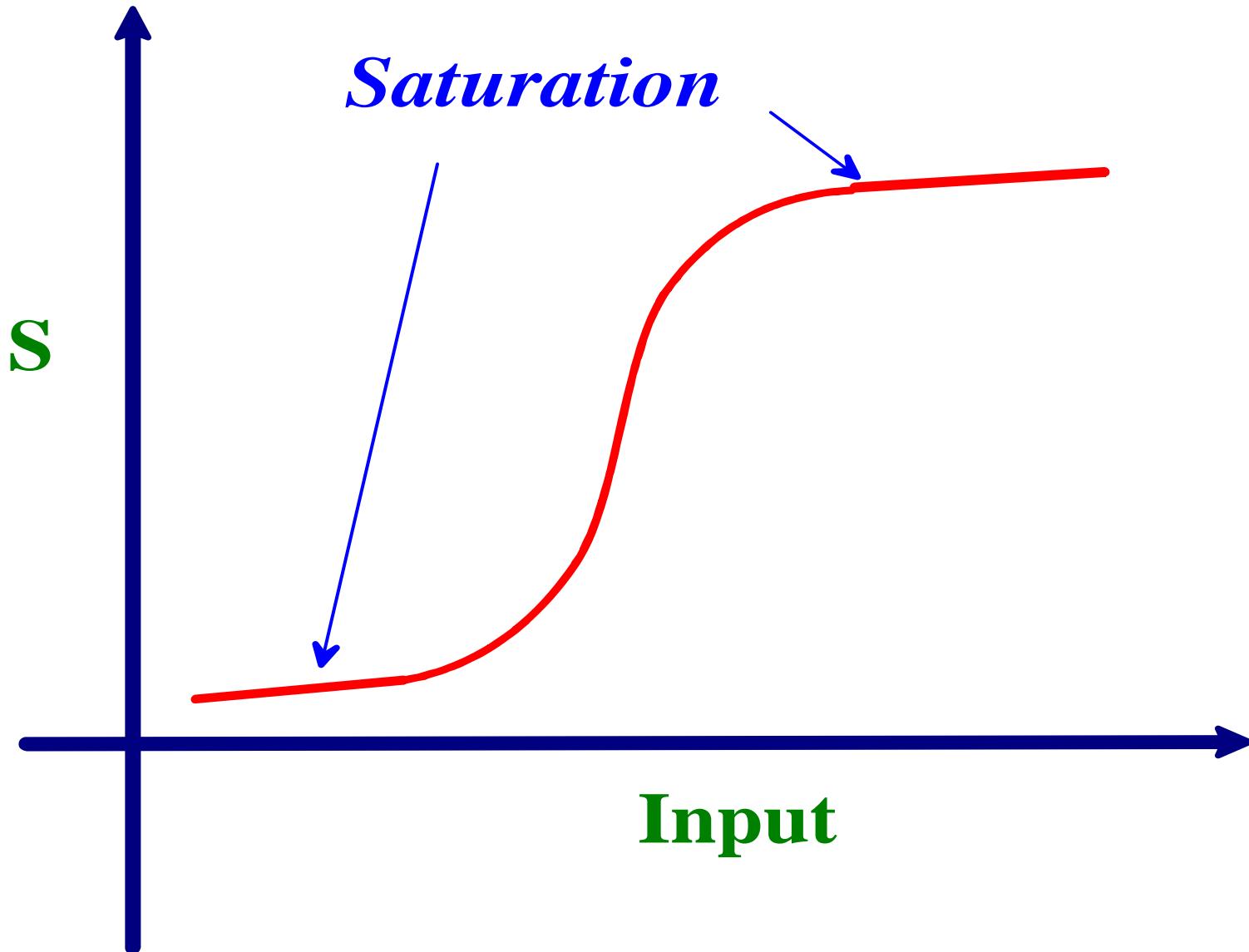
Normalized X



Variance Normalized Input

	<i>Feature 1</i>	<i>Feature 2</i>	<i>Feature k</i>
<i>Pattern 1</i>	$\frac{x_{11} - \mu_1}{\sigma_1}$		$\frac{x_{1k} - \mu_k}{\sigma_k}$
⋮	⋮	⋮	⋮	⋮
<i>Pattern n</i>	$\frac{x_{n1} - \mu_1}{\sigma_1}$			$\frac{x_{nk} - \mu_k}{\sigma_k}$

Network Saturation



Saturation Problems

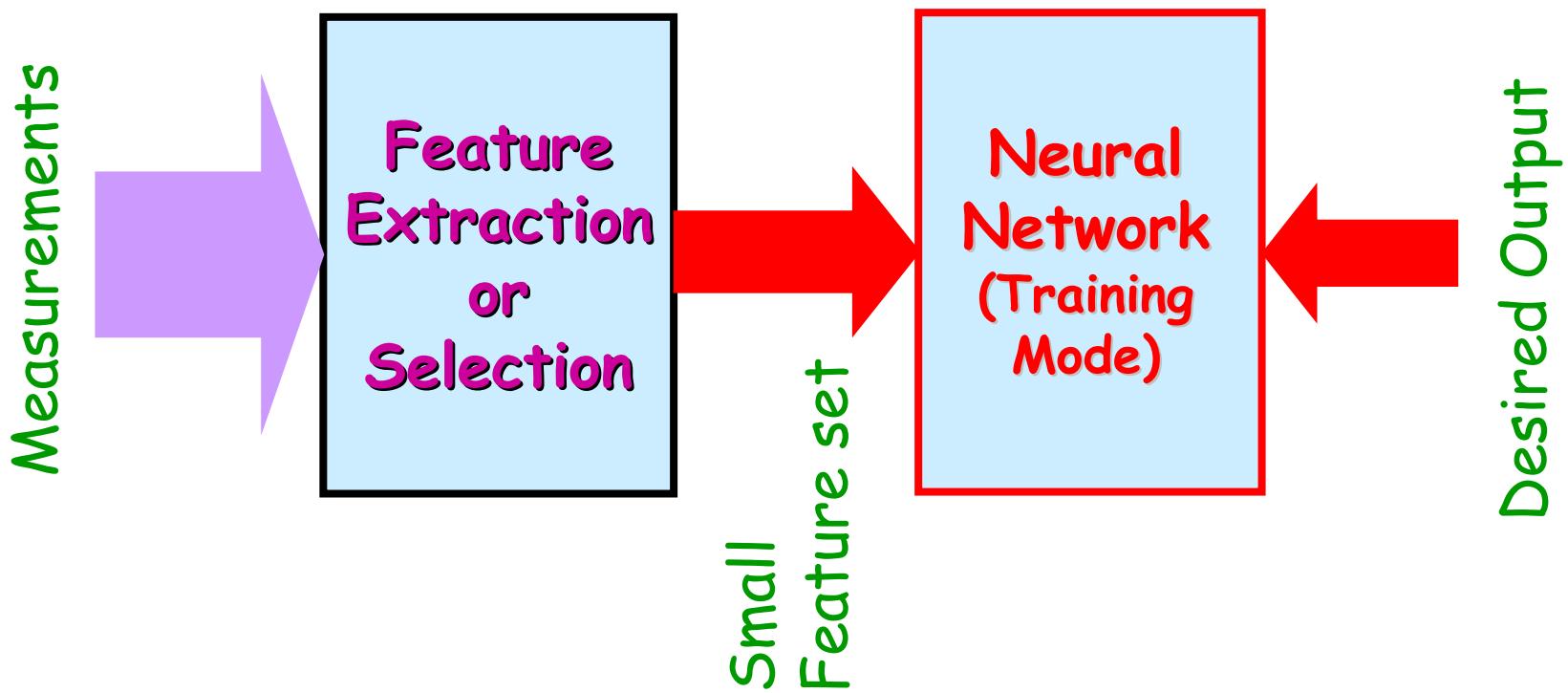
- Nonlinear functions reaches its limits
- A wide change in the input produce minimal change in the output
- With large number of saturated neurons, the NN can be paralyzed.
- If saturated, neurons must be randomly perturbed.

Feature Selection and Feature Extraction

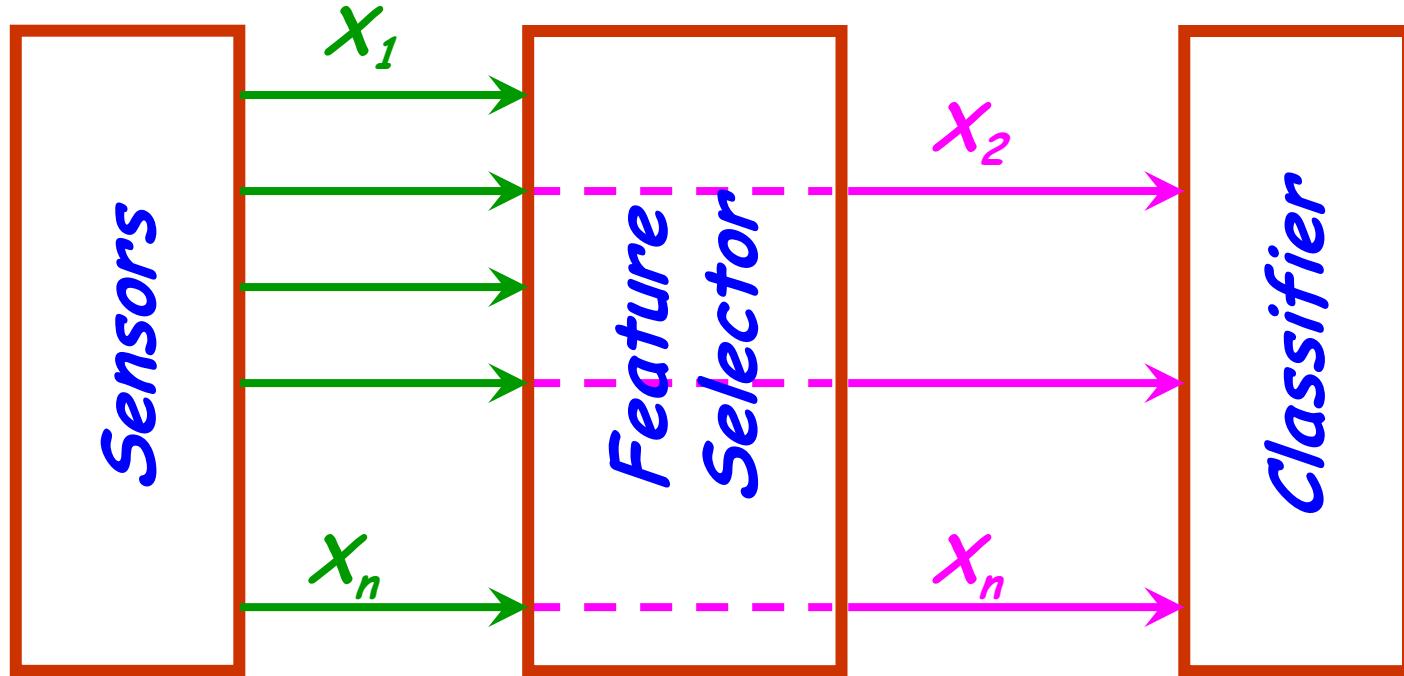
Feature Selection/Extraction

- *Eliminates curse of dimensionality.*
- *Enhances class separability.*
- *Reduces pattern dimension*
- *Maintains classification accuracy.*
- *Reduces training time*

Overall System Design: Training

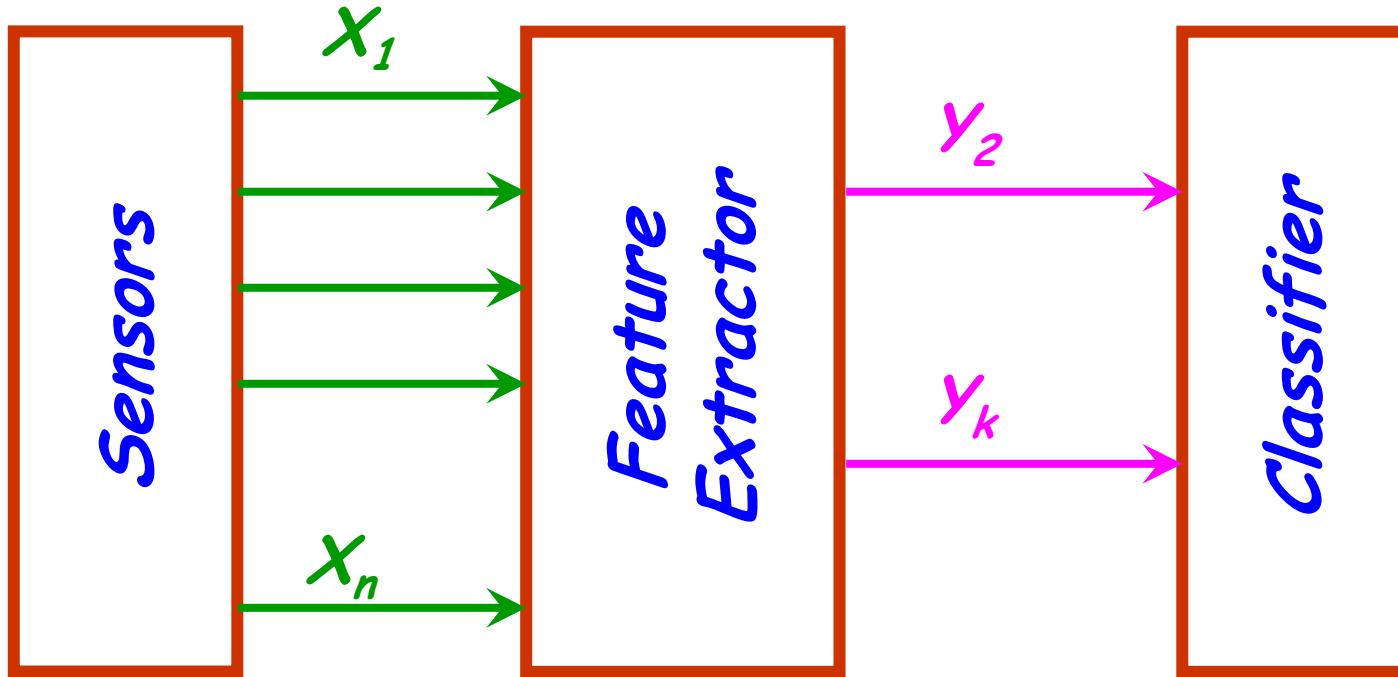


Feature Selection



*Most important features are selected
Techniques: Fisher Discriminate, Genetic
Algorithm, Particle Swarm Optimization*

Feature Extraction



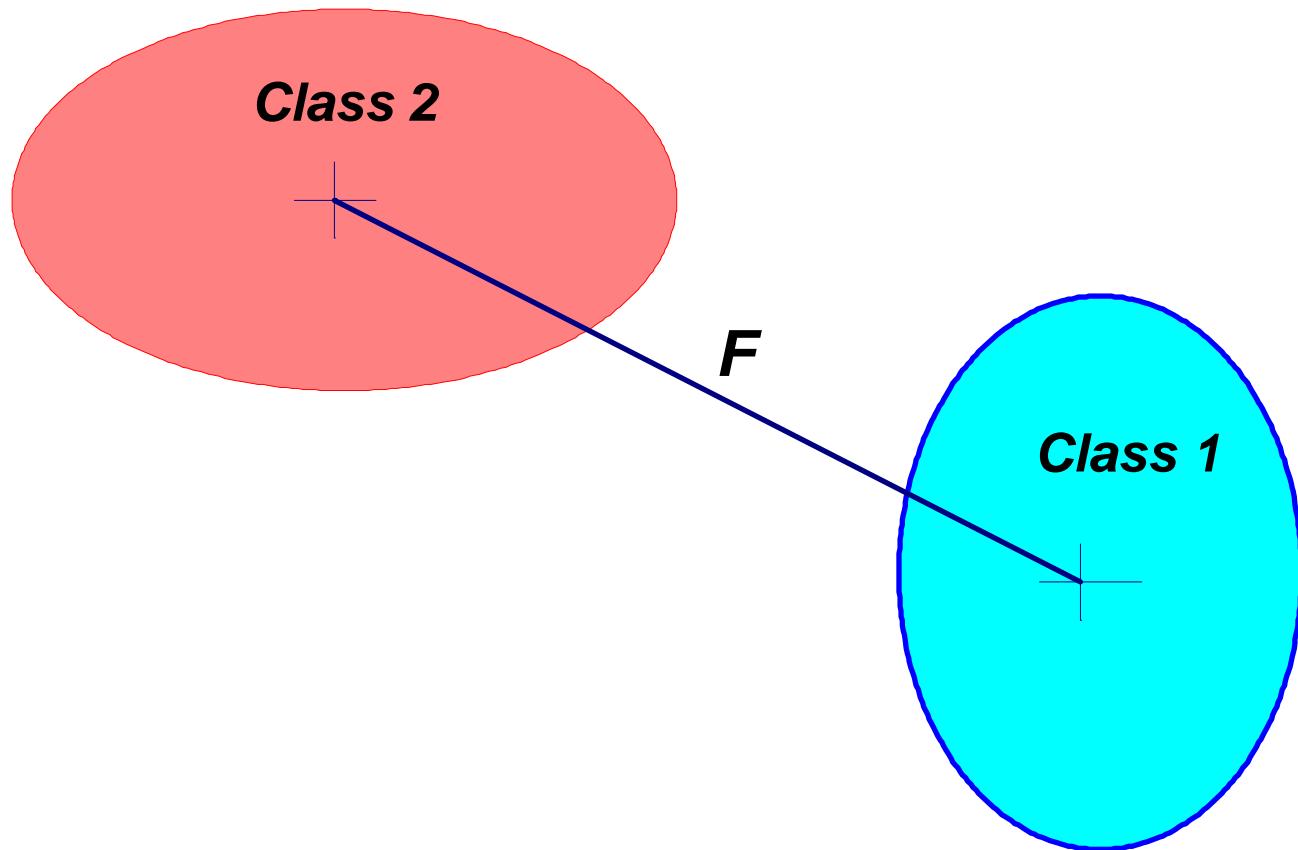
All features are combined to form a new reduced set of features

Techniques: Principal components, NN

Feature Selection/Extraction

- Class mean Selection (Fisher Selection)
- Karhunen-Loe`ve expansion
- Encoder

Fisher Classifier



Karhunen-Loeve expansion

- *Original pattern:*

$$[x_{i1} \ x_{i2} \ \dots \ x_{in}]^T; \quad i=1,2,\dots,M$$

- *Reduced pattern:*

$$[y_{i1} \ y_{i2} \ \dots \ y_{id}]^T \quad d \ll n$$

Karhunen-Loeve expansion

Original pattern of one class

$$X_i = [x_{i1} \dots x_{in}]^T, \quad i = 1, 2, \dots, M$$

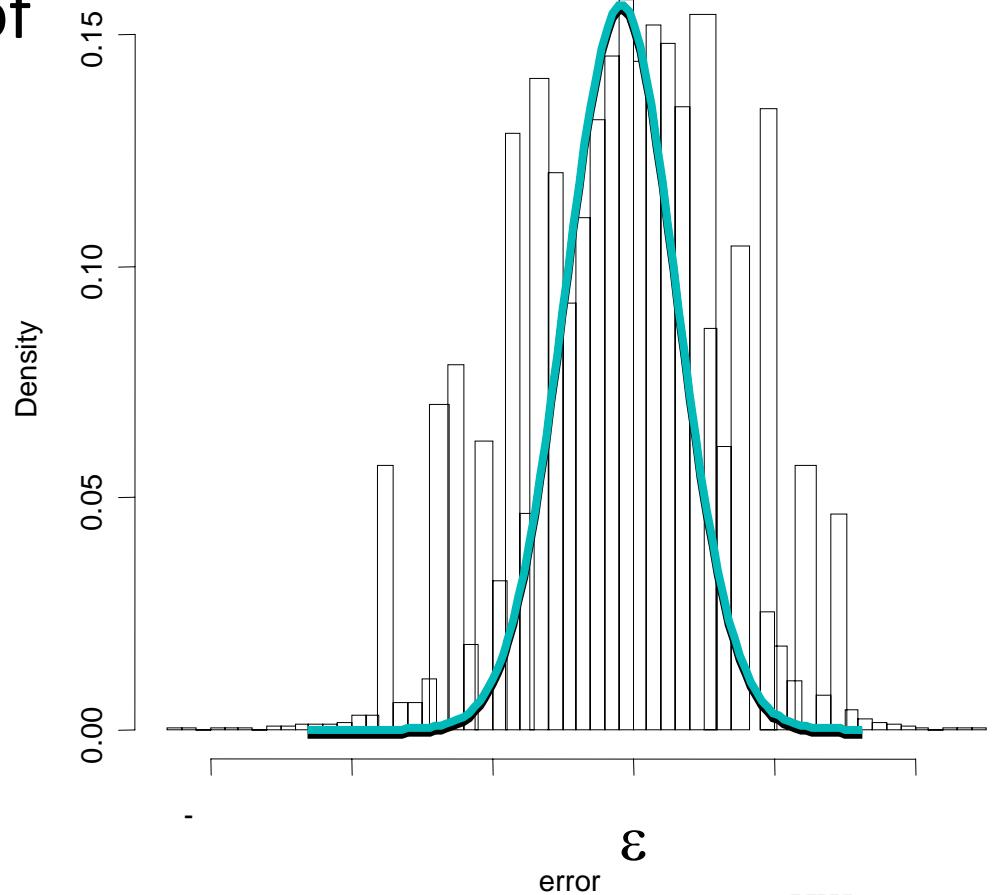
Pattern Mapping $X_i = \sum_{j=1}^n y_{ij} \Phi_j$

Φ is orthonormal function, y_{ij} is feature variable set

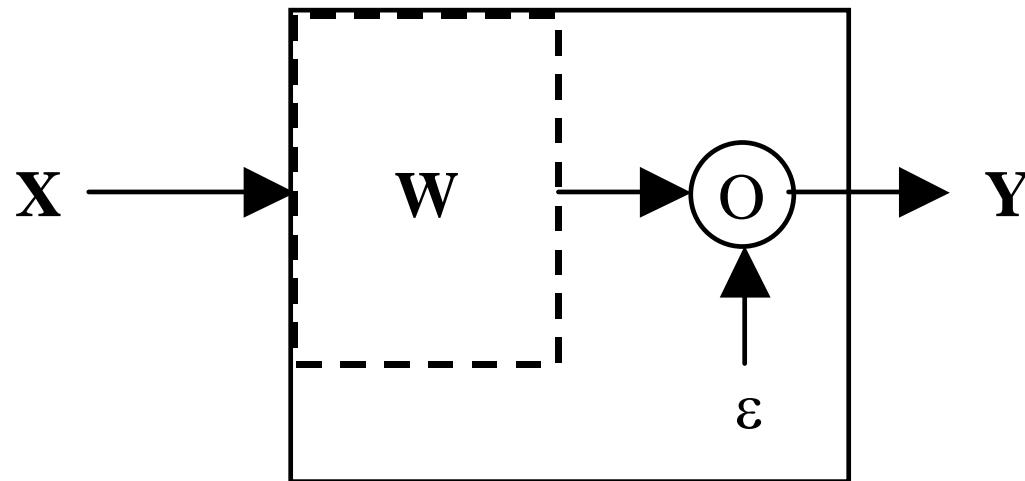
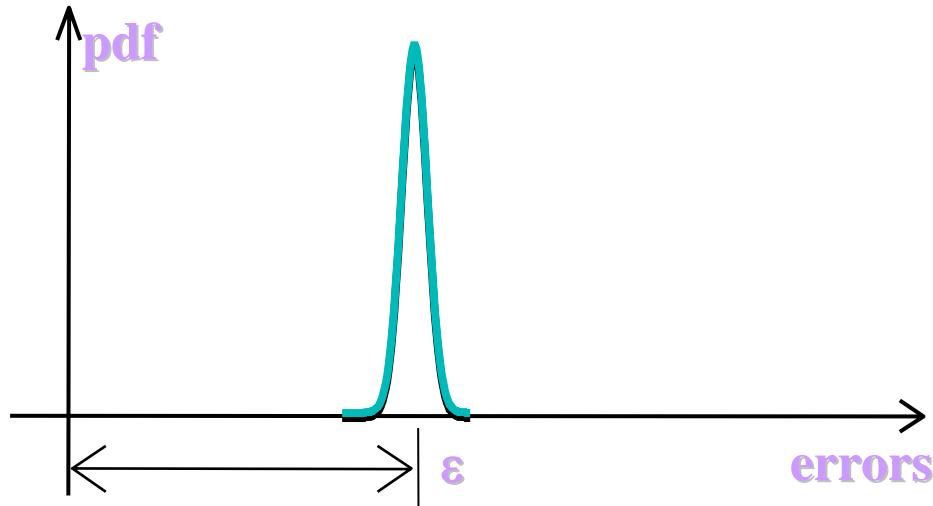
$$\Phi_i^T \Phi_j = \begin{cases} 1 & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases}$$

Non Gaussian Error

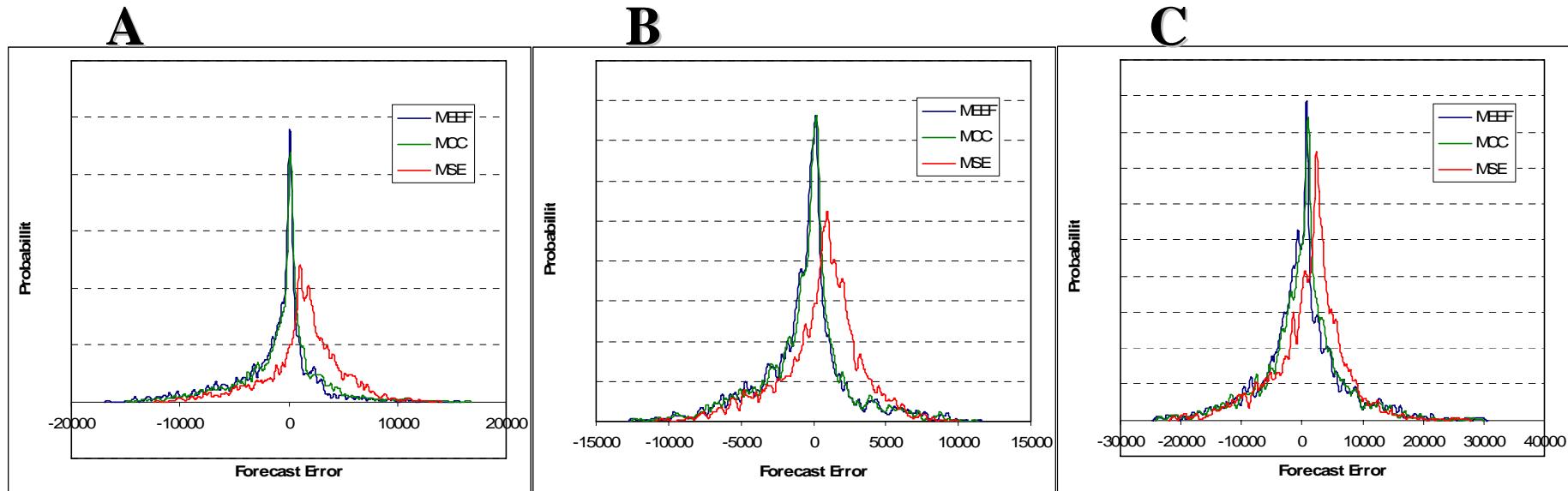
- The error distribution of wind speed (wind power) predictions from a NWM model does not satisfy the Kolmogorov-Smirnov test for Gaussianity
- Entropy Training addresses this problem



Entropy Training (Shannon, Renyi)



Comparing error pdf for all parks

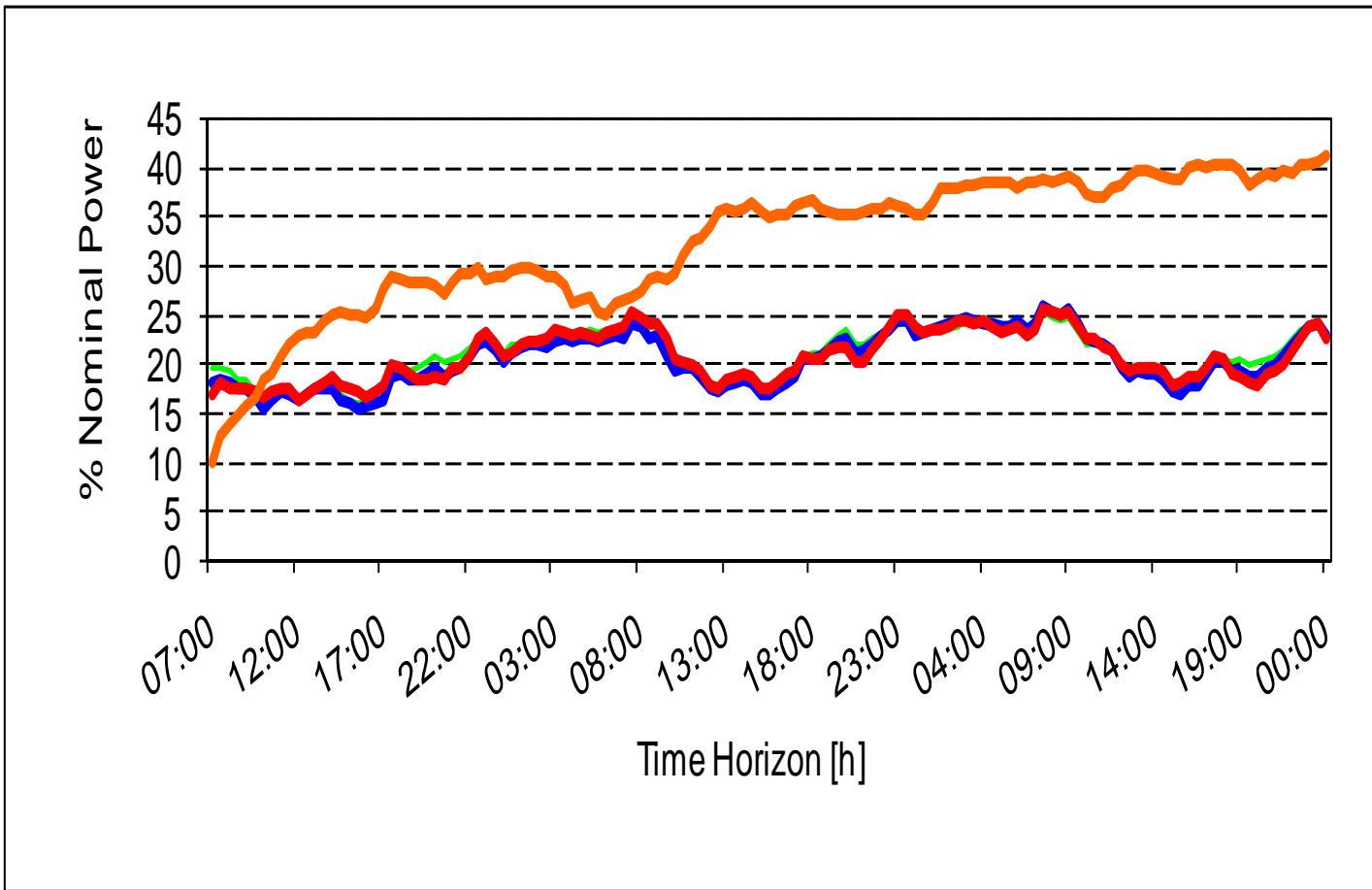


Courtesy of Professor Vladimiro Miranda, INESC, Portugal



Comparing NMAE

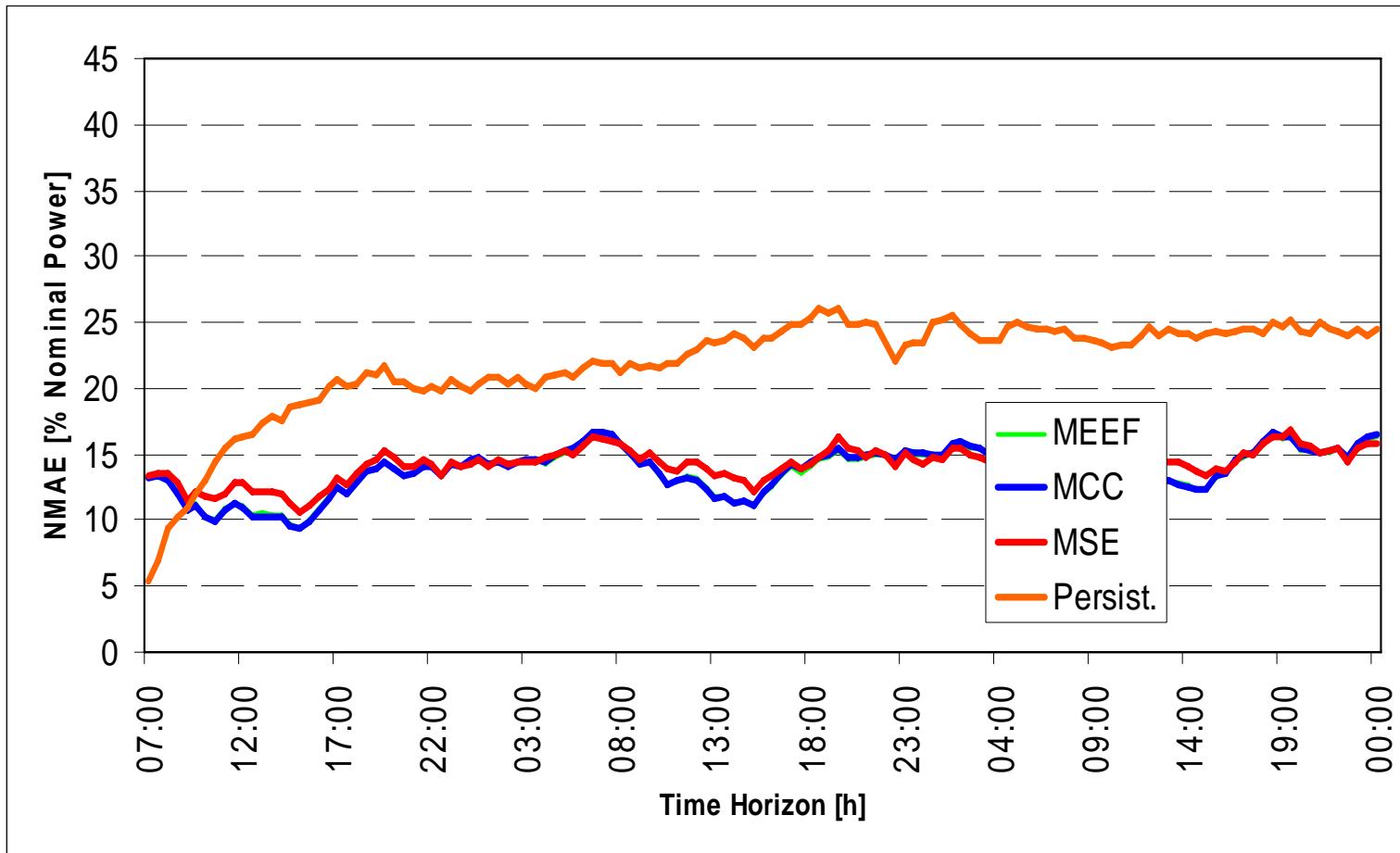
A



Courtesy of Professor Vladimiro Miranda, INESC, Portugal

Comparing NMAE

B

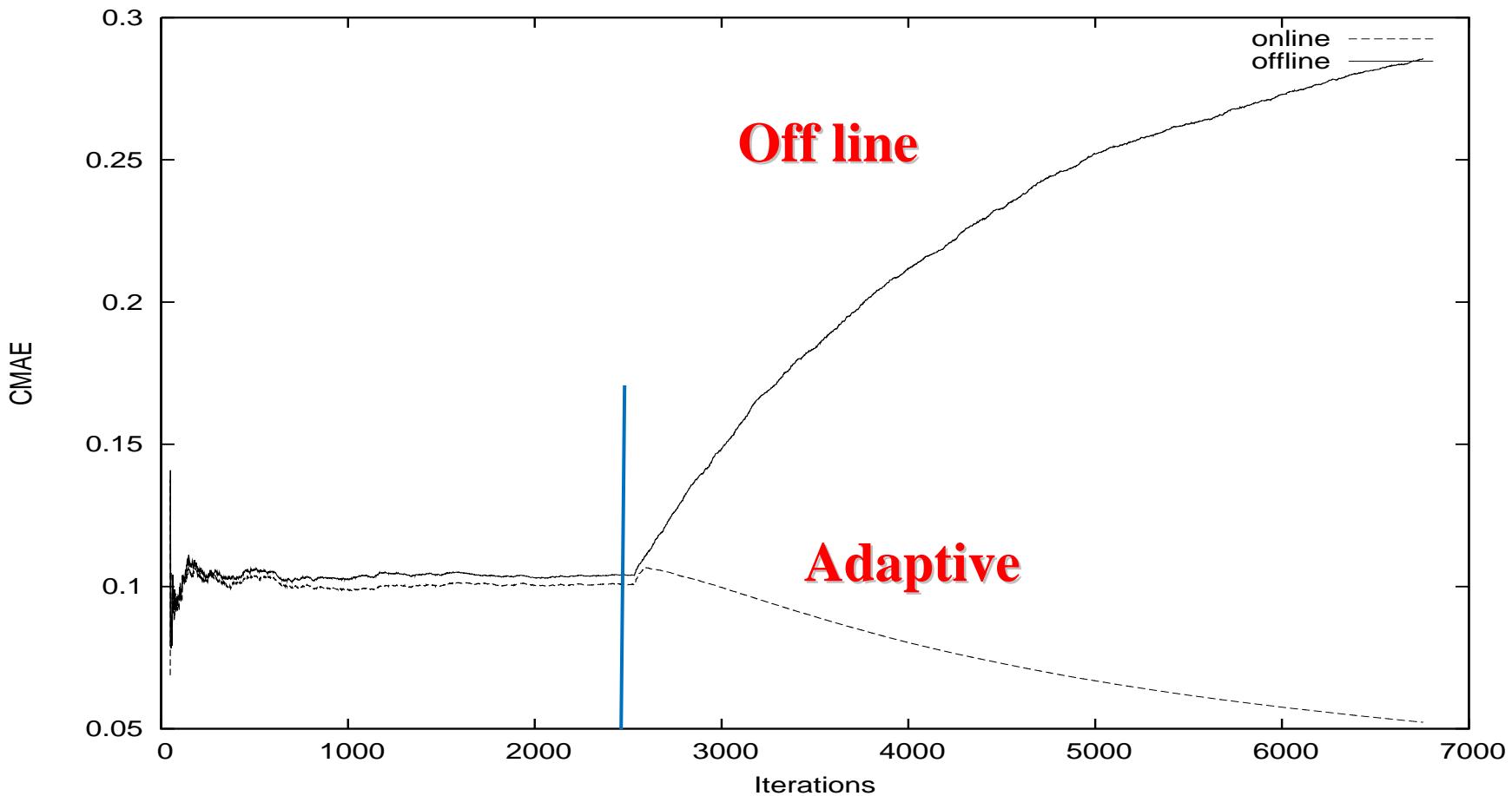


Courtesy of Professor Vladimiro Miranda, INESC, Portugal

ADAPTIVE FORECASTING

- For dynamically varying systems with/without large data sets (Load forecasting, security, etc.)
- Weights are automatically adjusted based on new data
- Effect of old and invalid patterns (data) are eventually and automatically deleted (forgotten)
- Perturbation in the NN weights are restricted to chosen boundaries
- Adaptive training does not drift

Adaptive Training



Conclusions

- NN can be treated as a black-box
- Least-squares is not the best choice for NN training
- Adaptive training is more accurate than off-line training

*Thank you for your
attention*